AppleScript-ohje

Perusteet

Tietoja AppleScriptistä

Kun käytät Macia, törmäät tehtäviin, jotka teet kerta toisensa jälkeen. AppleScriptillä voit kirjoittaa skriptejä ja ohjelmia, joilla voidaan automatisoida toistuvia tehtäviä.

AppleScript on tehokas ja monipuolinen skriptauskieli, joka on sisäänrakennettu OS X:ään. AppleScriptin avulla voit luoda oikoteitä, automatisoida toistuvia toimintoja tai jopa luoda muokattuja ohjelmia, jotka säästävät sinulta paljon aikaa. AppleScriptillä voit ohjata skriptattavissa olevia ohjelmia.

AppleScript-skripti koostuu englannin kielen kaltaisista komennoista, jotka suorittavat objekteille toimintoja. Esimerkkinä toiminnoista mainittakoon open, close ja get name. Esimerkkejä objekteista ovat ohjelma, ikkuna tai asetus.

Skripti voi ohjata yhtä tai useampaa ohjelmaa. Skriptattavissa olevassa ohjelmassa on määriteltävä skriptissä käytettävät termit. OS X:ssä on monia skriptattavia ohjelmia.

Skriptit voivat hallita ohjelmia verkon yli, joten voit automatisoida useamman kuin yhden tietokoneen toimintoja.

Olet itse asiassa käyttänyt AppleScriptia tietämättäsi. Monet käyttöjärjestelmän ja ohjelmien toiminnoista ovat todellisuudessa AppleScript-skriptejä, joita suoritetaan ohjelman sisältä käsin. Esimerkiksi, kun osoitat tässä ohjejärjestelmässä linkkiä, joka avaa ohjelman tai tiedoston, kyseessä on AppleScript-skripti, joka suorittaa toiminnon.

TÄHÄN LIITTYVÄT AIHEET

Kuinka AppleScript toimii
Tietoja AppleScript-editorista
Tietoja AppleScript-kielestä
AppleScript-resurssit internetissä
AppleScript Studion kehitysympäristö

Perusteet

Tietoja AppleScript-editorista

AppleScript-editori on helppokäyttöinen skriptausohjelma, joka voi lukea, kirjoittaa, nauhoittaa ja tallentaa AppleScript-skriptejä.

Tässä on joitakin AppleScript-editorin ominaisuuksia:

- Yhdistetty, helppokäyttöinen käyttöliittymä
- Tekstin etsiminen ja korvaaminen skripteissä
- Pitkien skriptilauseiden automaattinen rivitys
- Monitasoinen peruminen
- Kontekstivalikkoskriptit nopeampaa skriptien kirjoittamista varten

- Tukee ohjelmapalveluita
- Mahdollistaa ohjelmaskriptien käyttämisen kirjastopaletin avulla
- Sisältää Skriptiapurin, joka auttaa täydentämään lauseita
- Tarjoaa tuloshistorian ja tapahtumalokin historian, jotta voit tutkia, mitä skriptissä tapahtuu

TÄHÄN LIITTYVÄT AIHEET

Tietoja AppleScript-kielestä
AppleScript Studion kehitysympäristö

Perusteet

AppleScript-skriptien luominen

Sinun ei tarvitse tietää ohjelmoinnista yhtään mitään voidaksesi hyötyä AppleScriptista. Sinun ei edes tarvitse kirjoittaa yhtään skriptiä. OS X:n mukana tulee satoja hyödyllisiä AppleScript-skriptejä, joita voit käyttää, ja verkossa on saatavilla tuhansia lisää.

Jos haluat tutustua asiaan tarkemmin, voit mukauttaa olemassa olevia skriptejä omiin tarpeisiisi pienillä muutoksilla.

Saavutat luonnollisesti suurimman edun luomalla itse skriptejä. Opettelemalla kirjoittamaan skriptejä voit luoda hyödyllisiä työkaluja ja aikaasäästäviä automaatioita, jotka tekevät työstäsi tuottavampaa.

Voit muokata skriptejä tai luoda omia AppleScript-editorilla, joka tarjoaa yksinkertaisen ympäristön skriptien kirjoittamiseen. Kaikki AppleScript-ohjeen esimerkit on tehty AppleScript-editorilla.

- Avaa AppleScript-editori.

  Kirjoita seuraava teksti AppleScript-editorin ikkunaan.

  ```
  say "Welcome to AppleScript!"
  ```

  Osoita Suorita.

TÄHÄN LIITTYVÄT AIHEET

Tietoja AppleScript-editorista
AppleScript Studion kehitysympäristö

Perusteet

Kuinka AppleScript toimii

AppleScript toimii lähettämällä viestejä, joita kutsutaan Apple event -komennoiksi. Kun suoritat skriptin, AppleScript lähettää Apple event -komennot järjestelmään tai ohjelmiin, jotka suorittavat komennot.

Useissa tapauksissa ohjelmat tai järjestelmä tuottavat tuloksen AppleScriptiin. Esimerkiksi, jos skripti lähettää ohjelmalle pyynnön saadakseen objektin nimen, ohjelma palauttaa nimen. Jos komentoa ei voida suorittaa, ohjelma tai AppleScript voi palauttaa virheilmoituksen. Palautetut tulokset näkyvät AppleScript-editorissa.

Yleisesti ottaen sinun ei tarvitse tuntea Apple event -tapahtumien toimintaperiaatteita. AppleScript tekee tämän kaiken

taustalla. Sinun tarvitsee vain kirjoittaa ja suorittaa skriptit.

TÄHÄN LIITTYVÄT AIHEET

Tietoja AppleScriptistä
Tietoja AppleScript-kielestä

Tietoja sovelmista

Skriptiä, joka tallennetaan ohjelmana, kutsutaan nimellä AppleScript-ohjelma tai sovelma. Sovelmat suoritetaan automaattisesti avaamisen jälkeen ja ne suorittavat skriptin komennot. Sovelmat voivat olla pieniä toimintoja, jotka sisältävät vain pari komentoa tai monimutkaisia, tuhansista riveistä koostuvia ohjelmia.

Sovelmat voidaan tallentaa muokattavina tai pelkästään suoritettavina. Muokattava sovelma sisältää edelleen skriptin tekstin, joten sitä voidaan muokata ja kääntää uudelleen milloin tahansa. Jos haluat muokata tai muuttaa sovelmaa, vedä se AppleScript-editorin kuvakkeen päälle. Kun tallennat skriptin pelkästään suoritettavana, muokattava tekstiosuus poistetaan eikä tiedostoa voida enää muokata tai kääntää uudelleen.

Jos haluat muokata sovelmaa, joka tallennettiin muokattavassa muodossa, avaa se AppleScript-editorissa tai muussa AppleScriptien muokkausohjelmassa. Sovelmat vaativat ”on run” -käsittelijän, joka alkaa suorittaa skriptiä, kun sovelma avataan.

- Tässä esimerkissä on ”on run” -käsittelijä.

```
on run
        --first command
        --second command
end run
```

TÄHÄN LIITTYVÄT AIHEET

Tietoja dropleteista
AppleScript-tiedostotyypit

Tietoja käännetyistä skripteistä

Käännetty skripti on muokattavassa muodossa oleva AppleScript-tiedosto, jota voit lukea ja muokata AppleScript-editorissa tai muussa AppleScriptien muokkausohjelmassa.

Kun avaat käännetyn skriptin, se tulee näkyviin AppleScript-editorissa, mutta ei suorita automaattisesti mitään komentoja. Tämä mahdollistaa minkä tahansa muutoksien tekemisen skriptiin ilman sen suorittamista. Sitten voit kääntää skriptin, joka lukee ja muuntaa kaiken tekstin AppleScript-koodiksi. Käännetty skripti voidaan suorittaa. Käännetyn skriptin tiedostotarkenne on .scpt.

Kun tallennat tiedoston käännettynä skriptinä, teksti säilyy edelleenkin tiedostossa koodin ohella. Voit näin ollen muokata käännettyä skriptiä uudelleen koska tahansa.

TÄHÄN LIITTYVÄT AIHEET

Tietoja sovelmista
Tietoja dropleteista
AppleScript-tiedostotyypit

Perusteet

Tietoja dropleteista

Dropletti on AppleScript-skripti, joka suoritetaan, kun vedät tiedostoja tai kansioita sen kuvakkeen päälle. Skripti suorittaa toimintoja tiedostoissa tai kansioissa.

Kuten sovelmatkin, dropletti voidaan tallentaa muokattavana tai pelkästään suoritettavana. Muokattava dropletti sisältää edelleen skriptin tekstin, joten sitä voidaan muokata ja kääntää uudelleen milloin tahansa. Jos haluat muokata droplettia, vedä se Skriptieditori-kuvakkeen päälle. Kun tallennat skriptin pelkästään suoritettavana, muokattava tekstiosuus poistetaan eikä tiedostoa voida enää muokata tai kääntää uudelleen.

Dropletit vaativat ”on open” -käsittelijän, joka alkaa suorittaa skriptiä, kun dropletti avataan. ”On open” -käsittelijässä on parametri, joka sisältää luettelon dropletin päälle vedetyistä kohteista. Osa droplet-skriptiä tutkii luettelon ja toinen osa käsittelee kohteita. Ellei droplettia ole suunniteltu käsittelemään vain yhtä kohdetta, siinä tulisi olla toistosilmukka, joka käy luettelon kohteet läpi ja suorittaa toimintoja kullekin kohteelle vuorollaan.

- Tämän esimerkin dropletissa on ”on open” -käsittelijä ja parametriluettelo.

```
on open (theItemsDropped)
        --first command
        --second command
end open
```

TÄHÄN LIITTYVÄT AIHEET
Tietoja sovelmista
Tietoja käännetyistä skripteistä
AppleScript-tiedostotyypit

Perusteet

AppleScript-sanakirjojen käyttäminen

Voit selvittää, mitä komentoja ja objekteja voit käyttää ohjelman kanssa, ohjelman sanakirjasta. Sanakirja sisältää joukon termejä, jotka ohjelma ymmärtää.

Voit katsella ohjelman sanakirjaa AppleScript-editorissa valitsemalla ohjelman Kirjasto-ikkunassa tai vetämällä ohjelman kuvakkeen AppleScript-editorin kuvakkeen päälle.

Sanakirjan termit on järjestetty ryhmiin, joita kutsutaan paketeiksi. Komennot ja objektit näytetään lihavoidulla tekstillä.

1. Valitse AppleScript-editorissa Ikkuna > Kirjasto.
2. Kaksoisosoita ohjelman nimeä Kirjasto-ikkunassa.
3. Valitse kokoelma.

Kokoelman termit ja niiden kuvaukset tulevat näkyviin.

AppleScript-kieli on järjestetty eri kategorioihin:

- Kokoelmat: Toisiinsa liittyvien termien ryhmät. AppleScript-editorin sanakirjassa kokoelmat näkyvät ensimmäisessä sarakkeessa ja ne on merkitty oranssilla S-kirjaimella.
- Komennot: pyytävät toimintaa tai tulosta. Komennot, joita kutsutaan myös "tapahtumiksi", näkyvät toisessa sarakkeessa ja ne on merkitty sinisellä C-kirjaimella.
- Luokat: Skriptattavat objektit. Luokkien kuvauksissa luetteloidaan luokassa olevat ominaisuudet ja elementit. Luokat näkyvät toisessa sarakkeessa ja ne on merkitty violetilla C-kirjaimella.
- Ominaisuudet: Arvoja sisältävät objektit. Ominaisuudet näkyvät kolmannessa sarakkeessa ja ne on merkitty violetilla P-kirjaimella.
- Elementit: Muihin objekteihin sisältyviä objekteja. Esimerkiksi kappale voi sisältää monia sanoja. Elementit voivat näkyä kolmannessa sarakkeessa ja ne on merkitty oranssilla E-kirjaimella.

TÄHÄN LIITTYVÄT AIHEET

Luokkien määritelmät
Komentojen määritelmät
AppleScriptin käyttäminen ohjelmien hallitsemiseen

AVAA

AppleScript Language Guide: Commands Reference
AppleScript Language Guide: Class Reference

Perusteet

Ongelmien ratkaiseminen

Jos sinulla on ongelma AppleScriptin kanssa, nämä ohjeet saattavat auttaa ratkaisemaan ongelman.

Jos näkyviin tulee virhe, kun käyttäjä osoittaa Valitse luettelosta -valintaikkunassa Kumoa

Jos kansiotoiminto suoritetaan odottamattomasti

Jos skripti ei tunnista levyä

Peruskomennot

Luokkien määritelmät

Luokkamääritelmä kuvaa skriptattavan objektin ominaisuudet ja elementit. Luokkaa voidaan verrata lauseen substantiiviin ja luokkamääritelmää sanakirjassa olevaan sanan selitykseen.

Yleensä luokkamääritelmä AppleScript-sanakirjassa alkaa lihavoidulla termillä, jota seuraa samalla rivillä oleva lyhyt kuvaus termistä.

Tätä seuraa luettelo elementeistä ja objekteista, jotka kuuluvat skriptattavaan objektiin tai joista se koostuu. Kaikissa luokkamääritelmissä ei ole elementtiluetteloa, koska jotkin skriptattavat objektit sisältävät toisia skriptattavia objekteja ja

jotkut eivät.

Luokkamääritelmä päättyy skriptattavan objektin ominaisuuksien ja niiden arvojen luetteloon.

Ominaisuustietue alkaa ominaisuuden nimellä, jota seuraa:

- Kuvaus tietomuodosta, johon ominaisuuden arvo tallennetaan, kuten kokonaisluku (numero), kohta (kahden numeron luettelo), merkkijono (tekstimuodossa oleva data) tai looginen (tosi tai epätosi)
- Valinnainen tyypin osoitin [r/o] osoittaa, että ominaisuuden arvo on vain luku (”read-only”) eli se voidaan lukea, mutta ei muuttaa.
- Lyhyt kuvaus ominaisuudesta

Jos objektilla on monikkomuoto, voit viitata useampaan kuin yhteen objektiin.

Jos objektissa on luettelo elementeistä, se sisältää muita objekteja. Elementin nimi, jota voidaan käyttää skriptissä, näkyy lihavoituna. Normaali teksti kertoo, kuinka elementtiin viitataan.

Objekteilla voi olla ominaisuuksia. Ominaisuuksien nimet näkyvät lihavoituna. Normaali teksti kuvaa ominaisuuden sisältämää arvoa, esimerkiksi merkkijonoa. Ominaisuus voi olla vakio tai ilmentyä vakiojoukossa. Skripti voi hakea ominaisuuden arvon ja asettaa sen ellei ominaisuuden syntaksin jälkeen lue ”[r/o]” (kirjoitussuojattu).

TÄHÄN LIITTYVÄT AIHEET

Komentojen määritelmät
Objektien määritelmät
AppleScript-komennot
AppleScript-sanakirjojen käyttäminen

AVAA

AppleScript Language Guide: Class Reference

Komentojen määritelmät

Komennot ovat AppleScriptissä olevia termejä, jotka suorittavat toimintoja. Komennot näkyvät AppleScript-sanakirjassa keskimmäisessä sarakkeessa ja ne on merkitty sinisellä C-kirjaimella. Komennon määrittely sisältää lyhyen kuvauksen komennon toiminnosta ja komennon syntaksista. Syntaksi on tapa, jolla komento kirjoitetaan skriptissä. Syntaksissa saattaa olla useita osia.

Lihavoidut sanat ovat termejä, joita käytetään lauseissa. Nämä termit on kirjoitettava niin kuin ne on kirjoitettu sanakirjassa.

Sinisellä kirjoitetut sanat ovat antamaasi tietoa. Esimerkiksi määrittelemäsi objekti tai muuttuja.

Komennoissa on joskus valinnaisia osia, jotka näkyvät hakasulkeissa. Sinun ei tarvitse käyttää niitä lauseessa.

TÄHÄN LIITTYVÄT AIHEET

Luokkien määritelmät
AppleScript-sanakirjojen käyttäminen
Tapahtumien määritelmät
Objektien määritelmät

AVAA

Peruskomennot

Tapahtumien määritelmät

Tapahtumamääritelmä kuvaa verbiä tai komentoa, jota käytetään skriptattavan objektin toiminnon käyttämiseen. Tapahtumat ovat toimintoja tai verbejä, joita objektit voivat tehdä tai joihin ne voivat osallistua.

Tapahtumamääritelmä alkaa yleensä termillä, jota seuraa samalla rivillä oleva lyhyt määritelmä termistä.

Tätä seuraa komennon suorittamisessa käytettyjen parametrien luettelo. Vaihtoehtoiset parametrit ovat sulkeissa, pakolliset parametrit eivät ole.

Jokainen parametritietue alkaa parametrin nimellä, jota seuraa parametrin kanssa käytettävän tietomuodon kuvaus. Parametritietue päättyy lyhyeen kuvaukseen parametristä.

Komennon määrittely sisältää lyhyen kuvauksen (alleviivattu teksti) komennon toiminnosta ja sen syntaksista. Syntaksi on tapa, jolla komento kirjoitetaan skriptissä. Syntaksissa saattaa olla useita osia.

Lihavoidut sanat ovat termejä, joita käytetään lauseissa. Nämä termit on kirjoitettava niin kuin ne on kirjoitettu sanakirjassa.

Normaalilla tyylillä kirjoitetut sanat ovat antamaasi tietoa. Esimerkiksi määrittelemäsi objekti tai muuttuja.

Komennoissa on joskus valinnaisia osia, jotka näkyvät hakasulkeissa. Sinun ei tarvitse käyttää niitä lauseessa.

Komennot käsittelevät objekteja. Objektin määritelmä sisältää objektin tunnistamiseen tarvittavan syntaksin, mukaan lukien objektin nimen ja lyhyen kuvauksen objektista (alleviivattu teksti). Objektin luokka määrittelee tietyt objektien ominaisuudet, esim. objektin kanssa käytettävät komennot.

TÄHÄN LIITTYVÄT AIHEET

Tietokantatapahtumat
Luokkien määritelmät
Komentojen määritelmät
Objektien määritelmät
Kuinka AppleScript toimii

Peruskomennot

Objektien määritelmät

Komennot käsittelevät objekteja. Objektin määritelmä AppleScript-sanakirjassa näyttää objektin tunnistamiseen tarvittavan syntaksin, mukaan lukien sen nimen ja lyhyen kuvauksen (alleviivattu teksti). Objektin luokka määrittelee tietyt objektien ominaisuudet, esim. objektin kanssa käytettävät komennot.

Jos objektilla on monikkomuoto, voit viitata useampaan kuin yhteen objektiin.

Jos objektissa on luettelo elementeistä, se sisältää muita objekteja. Elementin nimi, jota voidaan käyttää skriptissä, näkyy lihavoituna. Normaali teksti kertoo, kuinka elementtiin viitataan.

Objekteilla voi olla ominaisuuksia. Ominaisuuksien nimet näkyvät sinisellä lihavoituna. Normaali sininen teksti kuvaa ominaisuuden sisältämää arvoa, esimerkiksi merkkijonoa. Ominaisuus voi olla vakio tai ilmentyä vakiojoukossa. Skripti voi hakea ominaisuuden arvon ja asettaa ominaisuuden arvon paitsi jos ominaisuuden syntaksin jälkeen on merkintä ”[r/o]”

(kirjoitussuojattu).

TÄHÄN LIITTYVÄT AIHEET

AppleScriptin käyttäminen ohjelmien hallitsemiseen
Komentojen määritelmät
Luokkien määritelmät
Objektin määrääminen
AppleScript-sanakirjojen käyttäminen

AVAA

AppleScript Language Guide: Class Reference
AppleScript Language Guide: Properties

Peruskomennot

If-lausekkeet

If-lause on komento, joka suoritetaan vain, jos tietyt ehdot täyttyvät. Jokainen if-lause sisältää totuusarvolausekkeen, joka arvioidaan joko todeksi tai epätodeksi. If-lauseen komennot suoritetaan vain, jos totuusarvolausekkeen arvo on tosi. If-lause päättyy end if -lauseeseen. Esimerkiksi:

```
if amountReceived > 100 then
        set memberStatus to "special"
end if
```

Jos yllä olevan lauseen totuusarvolauseke on tosi (amountReceived on suurempi kuin 100), if-lauseen komennot suoritetaan. Jos totuusarvolauseke on epätosi, komentoja ei suoriteta.

If-lause voidaan muodostaa myös else-lauseella. Tämä yhdistelmä tunnetaan nimellä if-then-else-lause. Esimerkiksi:

```
if amountReceived > 100 then
        set memberStatus to "special"
else
        set memberStatus to "general"
end if
```

Jos yllä olevan lauseen ensimmäinen totuusarvolause on tosi (amountReceived on suurempi kuin 100), lauseen ensimmäisen osan komennot suoritetaan. Jos ensimmäinen totuusarvolause on epätosi, lauseen toisen osan else-lauseen jäljessä olevat komennot suoritetaan.

Voit laittaa if-lauseita sisäkkäin useiden ehtojen käsittelemiseksi.

Lisätietoja AppleScriptin if-lauseista saat AppleScript Language Guide -oppaasta.

TÄHÄN LIITTYVÄT AIHEET

Repeat-lausekkeet
Komentojen määritelmät
Muuttujien asettaminen AppleScriptissä
Vakiot

AVAA

AppleScript Language Guide - If

Repeat-lausekkeet

Toistolauseilla voit suorittaa tehtäviä useita kertoja ehdoista riippuen. Toistolause jatkaa lauseryhmän suorittamista silmukassa, kunnes ehto on täytetty ja silmukasta poistutaan. Voit laittaa toistolauseita sisäkkäin erilaisiin ehtoihin reagoimiseksi.

AppleScriptissa on monia toistolauseita, joilla voidaan reagoida erilaisiin ehtoihin.

repeat (forever)

Toistaa jatkuvasti silmukassa. Poistu silmukasta exit-lauseella.

```
repeat
        -- statement
        -- statement
        if (x = 5) then exit repeat
end repeat
```

repeat (number) times

Toistaa määrätyn määrän kertoja.

```
repeat 5 times
        -- statement
        -- statement
end repeat
```

repeat while

Toistaa niin kauan kun määritetty ehto on tosi.

```
set x to 1
repeat while (x < 5)
        -- statement
        -- statement
        set x to (x + 1)
end repeat
```

repeat until

Toistaa kunnes määritetty ehto on tosi.

```
set x to 1
repeat until (x = 5)
        -- statement
        -- statement
        set x to (x + 1)
```

```
end repeat
```

repeat with (variable) from (startValue) to (stopValue)

Asettaa muuttujan arvoon startValue ja lisää muuttujan arvoa joka kerta, kun silmukka suoritetaan. Toisto loppuu, kun muuttuja saavuttaa lopetusarvon.

```
repeat with x from (5) to (25)
        -- statement
        -- statement
end repeat
```

repeat with (variable) in (list)

Asettaa muuttujan luettelon ensimmäiseen arvoon. Joka kerta kun silmukka suoritetaan, muuttuja asetetaan seuraavaan kohteeseen. Toisto loppuu, kun kaikki luettelon kohteet on käytetty.

```
set myList to {234, 342, 681, 874}
repeat with x in myList
        -- statement
        -- statement
end repeat
```

Lisätietoja AppleScriptin toistolauseista saat AppleScript Language Guide -oppaasta.

TÄHÄN LIITTYVÄT AIHEET

If-lausekkeet
Muuttujien asettaminen AppleScriptissä
Vakiot

AVAA

AppleScript Language Guide - repeat

Vakiot

Vakio on varattu sana AppleScriptissä ja sisältää esimääritellyn arvon. AppleScript sisältää lukuisia vakioita, joita voidaan käyttää tietojen hankkimiseen kohteista, vertailujen suorittamiseen tai aritmeettisten laskutoimitusten tekoon. Ohjelmat ja muut objektit voivat määrittää omat vakionsa. Nämä vakiot määritetään arvoluokkien määrityksissä.

Jos haluat nähdä vakion arvon, sijoita seuraava esimerkki AppleScript-editoriin ja osoita Suorita. Näet vakion arvon osoittamalla Tulos.

Myös tulos on vakio. Se on skriptivakio, joka viittaa komennon palauttamaan arvoon.

```
get the current date
```

Tässä on joitakin muita esimerkkejä vakioista:

- Aritmeettiset vakiot: pi

- Loogiset arvot: tosi, epätosi
- Päiväyksen ja ajan vakiot: minuutit, tunnit, päivät, viikot
- Merkkijonovakiot: rivinvaihto, välilyönti, sarkain
- Skriptivakiot: mikä tahansa, nykyinen ohjelma, se, minä, puuttuva arvo, minun, tulos

Lisätietoja AppleScriptin vakioista saat AppleScript Language Guide -oppaasta.

TÄHÄN LIITTYVÄT AIHEET

Muuttujien asettaminen AppleScriptissä
Objektien määritelmät
Luokkien määritelmät
Tapahtumien määritelmät

AVAA

AppleScript Language Guide - constants

Muuttujien asettaminen AppleScriptissä

Muuttuja on skriptissä luotu säiliö, jota käytetään arvon tallentamiseen. Muuttujaa luodessasi sinun on määritettävä sille nimi, jota kutsutaan myös tunnisteeksi. Kun skripti suoritetaan, voit sijoittaa muuttujaan arvon, käsitellä tai muokata sitä. Muuttujat kestävät vain niin kauan, kuin skriptejä suoritetaan. Kun skripti sulkeutuu, arvo menetetään.

- Näissä esimerkeissä kuvataan, miten yksilöllinen tunniste määritetään ja miten, sille annetaan arvo.

```
set myVar to "153"

set x to "Quarterly Reports"

set x to false
```

Jos haluat nähdä, miten arvot tallennetaan muuttujaan, sijoita nämä esimerkit AppleScript-editoriin ja osoita Suorita. Katso muuttujaan tallennettu arvo osoittamalla Tulos.

Lisätietoja AppleScriptin muuttujista saat AppleScript Language Guide -oppaasta.

TÄHÄN LIITTYVÄT AIHEET

Vakiot
If-lausekkeet
Repeat-lausekkeet

AVAA

AppleScript Language Guide - variables

Ohjelman sijainnin määrittäminen

OS X -ohjelmat ovat ”paketteja”. Paketti on hakemisto, joka sisältää kansiot ja tiedostot, joista ohjelma koostuu. Tämän vuoksi path to- ja info for -komennot saattavat antaa erilaisia tuloksia ohjelmille.

Jos käytät path to -komentoa ohjelmalle, polku päättyy yleensä ”.app”. Tiedostotarkenteita ei välttämättä näytetä Finderissa osana tiedostonimeä, mutta voit edelleen käyttää tarkenteita tiedoston tunnistamiseen skriptissä.

Jos käytät info for -komentoa tietojen saamiseen ohjelmasta, tuloksen Folder-ominaisuus on yleensä ”tosi”. Saat lisätietoja tiedostosta ja sen nimestä käyttämällä tuloksen lisäominaisuuksia, kuten extension hidden, name extension tai displayed name.

TÄHÄN LIITTYVÄT AIHEET

path to
info for
Kansioihin, tiedostoihin ja levyihin viittaaminen
choose application
Finderin skriptikomennot

Kansioihin, tiedostoihin ja levyihin viittaaminen

AppleScript mahdollistaa viittaamisen tiettyihin kansioihin, tiedostoihin ja levyihin nimellä ilman polkua. AppleScript tunnistaa kohteiden, kuten Dokumentit-kansion, työpöydän, kotikansion ja käynnistyslevyn, sijainnin.

- Tämä esimerkki näyttää, miten voit viitata erikoiskansioon käyttämällä sen nimeä skriptissä.

```
set x to the path to the desktop folder
```

TÄHÄN LIITTYVÄT AIHEET

Finderiin viittaaminen skripteissä
path to
info for
Finderin skriptikomennot

Finderiin viittaaminen skripteissä

Voit käyttää Finder-komentoja ohjaamalla komennot Finderille tell-lauseessa aivan kuten ohjaisit komentoja ohjelmalle.

- Tässä on esimerkki tell-lauseen käyttämisestä skriptissä. Tell-lause ohjaa ikkunoiden määrä -toiminnon Finder.appille.

```
tell application "Finder.app"
        return the number of windows
```

```
end tell
```

TÄHÄN LIITTYVÄT AIHEET

Finder
Objektin määrääminen
Kansioihin, tiedostoihin ja levyihin viittaaminen
Objektien määritelmät
Finderin skriptikomennot

AVAA

AppleScript Language Guide - Object specifiers

Finderin skriptikomennot

Näitä komentoja voit käyttää Finderissä:

activate

alias file

alias list

application file

clean up

close

column

Computer-object

container

copy

count

data size

delete

desktop window

desktop

disk

document file

duplicate

eject

empty trash

exists

file

file information

find

Finder

Finder window

folder

get

icon view options

information window

Internet location file

item

list view options

make

move

moving folder window for

open

package

path to

preferences

preferences window

print

quit

Kansioihin, tiedostoihin ja levyihin viittaaminen

Finderiin viittaaminen skripteissä

reveal

select

set

sort

Roskakori (trash)

update

window

Standardiskriptilaajennusten komennot

Skriptilaajennuksissa on monia hyödyllisiä komentoja, joita voit lisätä skripteihisi.

adding folder items to

beep

choose application

choose color

choose file

choose file name

choose folder

choose from list

choose URL

clipboard info

close access

closing folder window for

current date

delay

dialog reply

display

display dialog

do shell script

download

FTP item

get eof

info for

list disks

list folder

load script

mount volume

offset

open for access

open location

opening folder

path to

POSIX path

random number

read

round

run script

say

scripting components

set eof

set the clipboard to

set volume

store script

summarize

system attribute

time to GMT

upload

URL

write